

Cognitive and computer models of physical systems

S. CHANDRA

National Aerospace Laboratories, Bangalore, India

D. I. BLOCKLEY

Department of Civil Engineering, University of Bristol, Bristol, UK

(Received 10 October 1994, revised manuscript accepted 5 September 1995)

Models of physical systems range from those of initial individual cognition to mathematical representations on a computer which are accepted as the developed final models. It is conjectured that a formalization of the qualitative cognitive models will help us to understand how they are formed and will eventually help us to produce better computer models. The structure of these models would provide qualitative descriptions and explanations of behaviour which could be assimilated by non-specialists. It is argued that cognitive models should be produced with an awareness of the possible form of the final computer model. To illustrate this, a case study of the development of the cognitive and computer models of a naturally parallel physical process is presented. This early work is part of the broader goal of producing an appropriate computing environment through which various models and techniques are combined for producing explanations. A procedure for developing models from the primitive stage to computer implementation is suggested. Theories in cognitive science and research on mental models are briefly discussed.

© 1995 Academic Press Limited

1. Introduction

The forming of cognitive models (CM) of physical behaviour (e.g. water flowing in a pipe or the behaviour of a beam under loading) is an important topic in cognitive science. These models are qualitative and are usually referred to, in the terminology of cognitive science, as being primitive theories. Most physical behaviour models (PM) are quantitative, are based on physical laws, are expressed in mathematical form and are implemented on a computer. The development of CMs and their relationships with PMs merits careful examination.

The purpose of this paper is to examine how the early CMs are built and the nature of their relationships with the final PM models for three reasons. Firstly the cognitive modelling process is important in its own right. It appears to be based on analogical patterns, the understanding of which are important in the evolutionary development of all models. Secondly the structure of a CM is important for the final computer model PM if the latter is to be a faithful representation of the CM and able to provide explanations. The primitives within the CM will be used to describe qualitative behaviour which is derived quantitatively in the PM. Thirdly if PMs are

to be developed so that they can provide qualitative explanations of physical processes, within for example engineering design, then the language and vocabulary used should be that of the CM.

The success of common-sense reasoning about physical processes is a powerful reason to examine closely theories of qualitative physics. Models that can be built from primitives that have only local effects but which interact to form more complex global behaviour are potentially important. However the standard quantitative theories of physics expressed in mathematical form are so well established and useful that it is important to include them.

It will be argued that there is a need to find a formalism for the development of early cognitive models. This formalism should have an appropriate structure into which mathematical models can be embedded and quantitative data can be mapped into qualitative explanations. There is some support for this view from work on mental models (Gentner & Stevens, 1983; Staggers & Norrcio, 1993). There is a conjecture, for example, that human beings identify a set of functional objects and a topology in their minds which they can "run". Pylyshyn (1989) describes a functional architecture to model the structure of some physical processes especially if the computer is used to explain rather than calculate behaviour. De Kleer and Brown (1984) believe that human beings may use a kind of qualitative causal calculi to manipulate knowledge about the physical world. A concept of appropriate physics (Chandra, 1992) which incorporates various levels of representations, from the qualitative to the mathematical, may well emerge from the work described here. This will enable the development of levels of contextual explanations, both qualitative and quantitative, in a form understandable to the novice or the expert. Engineering designers routinely test the results of quantitative analyses against their own cognitive models to gain confidence in those results. On occasions when such results are counter-intuitive to the CM an interesting interaction will develop resulting in real learning either in the CM or PM or both.

The example of natural parallelism is used here to illustrate the process of development and implementation of CM and PM. Specifically a problem in structural dynamics is chosen since it is generally accepted that physical processes are composed of the aggregation of naturally parallel simpler processes but this natural parallelism is not reflected in the presently developed PM. Rather *ad hoc* task parallelism is identified and implemented. The work is motivated by the development of an IOPM (Interacting Objects Process Model—Section 6.0) which incorporates both practical and theoretical advances (see also Chandra, 1992, 1993, 1994).

2. Mental models

The term mental model is now widely accepted in human-computer interaction literature. However, other terms used to describe these models are cognitive models, conceptual models, component models or causal models. There is growing evidence that cognitive scientists and researchers in human-computer interaction are particularly interested in the differences between the models of experts and novices.

However, it is acknowledged that such research is difficult since the measurement of internal models is problematic (Staggers & Norcio, 1993).

Mental models are believed by many to be organized patterns or structures which may be represented as objects and relationships (Gentner & Stevens, 1983; Williams, Hollan & Stevens, 1983; Staggers & Norcio, 1993). Staggers and Norcio (1993) believe that objects in mental models are related to perceptual entities. Gentner and Stevens (1983) postulate that these models are formed by structuring analogies and metaphors, especially in unfamiliar domains. A mental model is regarded as a naturally evolving appropriate dynamic representation in the mind about any system or object in the real world (Norman, 1983). Mental models are the conceptual models that scientists and designers use to describe theories and artifacts. The operations at this level are not based on axioms and the notions are intuitive and not directly evident. Even at the root of scientific theories and axiomatic systems (mathematics) there is a layer that is primitive in the sense that it is not explicitly explained or justified within the system. It was recognized by Goethe (Zajonc, 1976; Gentner & Stevens, 1983) who believed that any scientific explanation begins with common-sense reasoning about disparate and isolated cases. According to Goethe, what then follows is a process of sifting through the cases, finding successively the more general and fundamental ones which serve as principles. Common concepts such as space, time, force, energy, and heat are intuitive at the highest level of the modelling process (Trainor & Wise, 1979). They have been accepted through the scientific process of discovery of similar inter subjective collective experiences and made dependable by measurement (Blockley, 1980).

2.1. AUTONOMOUS OBJECTS

Williams *et al.* (1983) consider that mental models are composed of a set of autonomous objects with an associated topology.

"Central to this conception of mental models is the notion of an autonomous object. An autonomous object is a mental object with an explicit representation of state, an explicit representation of topological connections to other objects, and a set of internal parameters. Associated with each autonomous object is a set of rules which modify its parameters and thus specify behaviour. Running a mental model corresponds to modifying the parameters of the model by propagating information using internal rules and specified topology."

Autonomous objects are regarded in this work as containing knowledge with variables that can be instantiated. The computational metaphor is object-oriented computer data structures. These objects can be regarded as abstractions that have definite boundaries. Williams regards the behaviour of autonomous objects as being governed strictly by internal rules reacting to internal parameter changes and to highly constrained external provocation.

Inherent in the concept of autonomous objects is an explicit representation of the "connectedness" of the objects. Objects interact with a limited number of other connected objects by passing changes in parameter values. Though objects can be regarded as opaque, they can be decomposed. This embedding results in a new

mental model, itself composed of autonomous objects of a given topology, which can be used to produce explanations of the initial higher level objects. The decomposition of these autonomous objects yields a multi-level hierarchical modelling of a system. In this work, it is believed that the computer representations must be based on the identification of objects which represent actual physical processes or entities whose behaviour on aggregation represent a portion of a physical process. However mental models can be incomplete, inaccurate and parsimonious. Individuals can over-generalize or over-map if they import existing models or use multiple mental models.

It seems that there is now a consensus emerging about several issues. Mental models are internal representations of systems in particular knowledge domains. These internal representations are formed through knowledge (instruction) or experience or a combination of the two and the resulting model is a private enterprise, an individual matter. How mental models are constructed is not known. Some authors, however think that humans construct models by using analogies or metaphors of past represented objects or interactions. In general, authors think models are composed of both concrete and abstract objects in a relational structure which emulates the actual situation. Humans have the ability to "run" models in their heads for both descriptive and causal hypothesis testing. In this way models may be modified and constructed. Mental models seem to be more *process* than outcome entities because of their dynamic and changing nature.

The role of vocabulary in explaining phenomena remains crucial. Different sciences, for example, explain phenomena in different ways. This is more than a superficial question of terminology. Scientists cannot choose a vocabulary entirely for its interest and convenience, though they can augment and refine the vocabulary to suit their needs. Explaining is a special activity the success of which depends on how the explanation relates to the way we describe the phenomenon to be explained. A typical way to explain phenomena in natural science is to cite nomological, or universal laws and relate them to descriptions of phenomena couched in the same vocabulary as that of the laws (or to descriptions of the phenomena defined in terms of this vocabulary). The thesis put forward by Pylyshyn (1989) is that some phenomena seem to demand explanations couched in a certain vocabulary. Phenomena can be described using a variety of terms that taxonomize the world in various ways. A hierarchy of relationships exists among such terms. At each level, important and interesting generalizations exist. The most efficient explanations are those related to the earliest CM. It is common to find, for example, that access to knowledge is highly context dependent. Subjects often have good access to problem-solving tactics if the problem is worded one way, but not when the problem is worded another way (Hayes & Simon, 1976). The concept of tacit knowledge—as a generalization and an extension of the everyday notion of knowledge (much as physicist's concept of energy is an extension of the everyday notion)—is one of the most powerful ideas to emerge from contemporary cognitive science, though much remains to be worked out regarding the details of its form and function. It is already clear, however, that tacit knowledge cannot be freely accessed or updated by every cognitive process within the organism, nor can it enter freely into any logically valid inference; contradictions are much too common for that to be true.

Pylyshyn (1989) states that cognitive science is founded on the assumption that there exists a natural set of generalizations that can be captured by using cognitive terms. To succeed, we must adopt the right taxonomy or descriptive vocabulary. Another equally important goal is to capture the generalizations in a manner that does more than merely tap our informal knowledge of the way people behave. We also want to be able to relate these generalizations to possible physically realizable mechanisms.

3. Computation and cognition

In this section it will be argued that the formal development of the model is through the identification of constraining parameters that fix the basic characteristics of the system.

Pylyshyn (1989) notes that if a programmed computer (or even a formally specified computational process) is to be a serious candidate as an explanatory model of mental processing, there should be an explicit account of the way the model relates to the domain of phenomena it is to explain. If the computational system is to be viewed as a model of the cognitive process, it must correspond to the mental process in detail. This essentially means that we have to build an early model of the structure of the processes we intend to simulate on the computer. This formal structure of the virtual machine is what Pylyshyn (1989) calls a functional architecture—thus representing the theoretical definition of, for example, the right level of specificity (or level of aggregation) at which to view mental processes.

Devices with different functional architectures cannot, in general, directly execute identical algorithms. Pylyshyn (1989) adds that in order to describe an algorithm so that it can be viewed as a cognitive process, we must present the algorithm in standard, or canonical form or notation. Typically this means formulating the algorithm as a program in some programming language; it might also include graphical presentation or even a natural language description. What is often overlooked is the extent to which the class of algorithms is conditioned by the assumptions made regarding the possible basic operations. These include how they interact, how operations are sequenced, what data structures are possible, and so on. Such assumptions are an intrinsic part of the descriptive formalism chosen, since it is that formalism which defines what Pylyshyn (1989) calls the “functional architecture” of the system.

The exploitation of the functional architecture of these computational systems is central to computer science. From the point of view of cognitive science, though, it is not only important that we chose the functional properties of the computational architecture but it is also important to use computational models in an explanatory mode rather than simply a performance mode.

Thus, a basic architecture of a formal model is required not simply because in this way we can get to the most primitive level of explanation, but rather because we can get a principled and constrained (and therefore, not *ad hoc*) model only if we first fix those properties which are basic characteristics of the system. Pylyshyn (1989) states that unless we set ourselves the goal of establishing the correct functional

architecture properly to constrain our models, we could well find ourselves in the position of having as many free parameters as we have independent observations.

Thus far a set of criteria for the development of the cognitive model for the particular problem have been established. These are firstly an object-oriented functional structures for the model. Secondly, the development of a formal vocabulary and terminology. Thirdly, a functional architecture that takes note of the computational architecture and fourthly, the need to use computational models in an explanatory mode.

4. Composability and locality

Physical systems have mainly been described in the form of models and mechanisms in three different forms of ontologies, component based, process based, and constraint based. Qualitative physics (Forbus, 1984; Kupiers, 1984; Forbus & Flankenhainer, 1990; Leitch, Weigand & Quek, 1990) deals with inferences based on common sense reasoning about processes. The underlying mechanism in process based ontology is the influence of one process over another; in the component based analogy, it is message passing; in the constraint based method, it is the satisfaction of constraints. In this work PM are process based with a number of independent processes whose behaviour is influenced by communication (message passing between neighbours) and satisfaction of constraints that in this particular case are physical laws and mathematical relations.

In qualitative physics, composability is a requirement which implies that the description of the process must be derivable from the structure of the system. In other words, a qualitative description of the typical behaviours of the system should be available from the structure of the system. The other requirement is that simple functional models can be composed to form structures to describe more complex situations. Locality, another requirement from qualitative physics follows from this. This implies that effects must propagate locally between components of a process and neighbouring processes. The term structure here refers to the components of the analysis, component behaviours and connections between components. The term behaviour refers to the observable changes in the state of the components through time.

5. Modelling requirements and procedures

The modelling requirements set out in the previous sections can now be summarized. In Section 2.1 it was argued that a multi-level hierarchical model is possible with computer objects which can represent processes or entities and whose aggregated behaviour represents a complex process. In Section 3 it was pointed out that the type of functional architecture that will be available must be considered before the development of the model if it is to serve as an explanatory vehicle for the cognitive model. In Section 4 it was argued that composability and locality are useful requirements to set at the cognitive stage for the development of an efficient

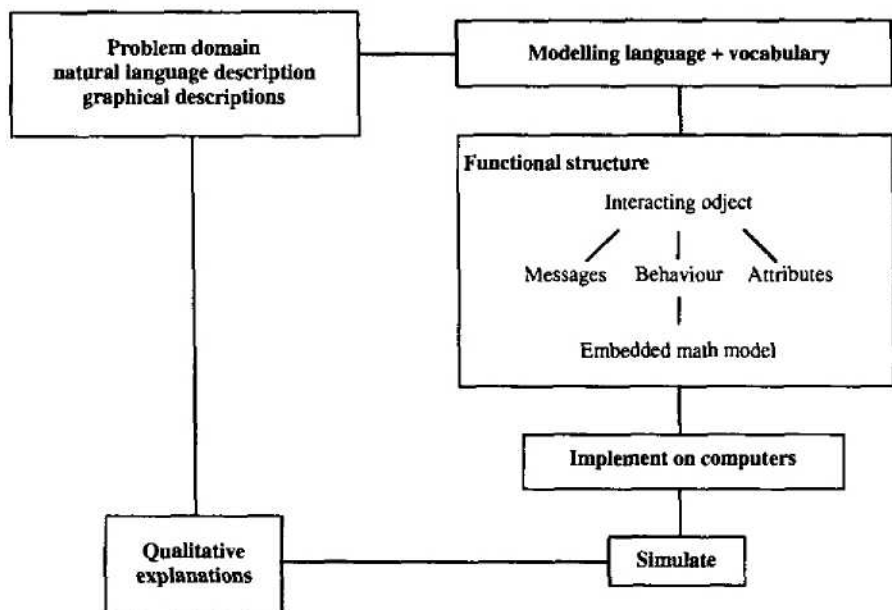


FIGURE 1. Modelling procedure.

model. Figure 1 sets out a modelling procedure that is conjectured as being more efficient for firstly understanding the evolutionary development of CM and secondly for using available and proven PM in an environment where the behaviour can be explained qualitatively in terms of the CM.

6. The Interacting Objects Process Model (IOPM)

A model for a specific physical system has been developed (Chandra 1992, 1993) which conforms to the ideas described earlier. Analogies are used to build the early mental models. A graphic representation aids the development of a natural language description. This description is then formalized by developing a Physical Process Modelling Language (PPML). This enables the development of a naturally parallel functional architecture. The physical system behaviour is considered to be the aggregation of processes (behaviours) which pass messages to each other. Each individual process is represented by a set of functional objects (components) and physical laws are the constraints that have to be satisfied.

6.1. THE PROBLEM DOMAIN

The problem domain is structural dynamics. The intention is to model the behaviour of a physical system after it is impacted or disturbed from its initial position by external action. The first requirement of modelling is description. Description attempts to create an image of behaviour and draws on the emergence of repetitive

patterns, order, relation and structure. Description leads to the synthesis of natural laws. With properties that appear to be related in some form and specified in some form of definiteness, it becomes possible to express the relation in precise language. Lindsay (1968) sees a physical law as a compact description of a routine of human experience. Space and time are notions essential for mental imagery of physical systems and Lindsay identified the concept of space in three forms: (1) the private space of each individual, (2) public or physical space, (3) conceptual or mathematical space. Leibnitz defined private space as the "order of co-existent sense impressions". Even when space is regarded as a private concept it is necessary to involve the idea of time. This can be identified in the same manner. Private space and time is inadequate for science since it is based on individual sensory perceptions. It has been essential to replace these with an abstraction of the mind where the relation of a physical object to an observer is determined by the properties of an ideal space. This enables measurement and an isotropic and homogeneous physical space.

Similarly the difficult concept of time serves as an independent variable which, along with definitions of space, enables changes in physical experience to be represented. The abstraction of space and time leads to the ability to measure change. The measurement is made on the attributes of the physical system under consideration. Rosen (1985) regards these attributes as "observables".

Reactive behaviour is assumed here to be a common part of all physical systems. For systems that involve more than one "physical" object like a train of masses and springs, information transfer between these objects is required. An obvious analogy is that with wave motion. Representation of behaviour either as waves or as particles is a classic duality in physics which has been reconciled at appropriate definitions (Feynman, 1965, Conrad, Home & Josephson, 1988). Burton (1968) uses the wave motion analogy as an information transfer mechanism. For example, if a disturbance is initiated at any point in a medium, the disturbance moves with a velocity. The regions that are further away from the point of disturbance "do not know" that a disturbance has been applied at that point in time. Burton used an analogy of traffic which frequently behaves like a vibratory system and explains that the rate of information flow that moves along the stream of traffic can be described in the form of a wave motion. The information received by the drivers of the cars results in reactions that are transmitted along the stream of traffic.

Analogically a physical system can be represented as a set of objects each of which received information about its neighbours and reacts to the messages. We will assume that information transfer is via a wave motion mechanism which then operates in the following way:

- (1) a disturbance travels at a certain velocity which depends on the physical properties of the medium;
- (2) objects react to an external action from a meta system or to a message from a neighbour;
- (3) all objects behave independently at a given instant of time after receiving messages.

The velocity of propagation of sound in a solid media can be measured and used to set the time required for a message to travel from one object to another.

6.2. QUALITATIVE MODELLING OF A SIMPLE PHYSICAL SYSTEM

The concept of the IOPM presented in this paper can be described by developing the early cognitive models about a simple physical system which is naturally parallel. This natural parallelism implies that spatially unconnected events will not influence each other at a particular instant of time—but will do so later as messages are transmitted through the system. In other words complex behaviour can be represented by the aggregation of relatively simple processes over a period of time. This is illustrated by describing a simple physical system in such a way that the natural parallelism is utilized at an early stage in the building of the model. The arguments developed earlier in this paper are then used to develop a modelling language, identify a set of functional objects and develop mathematical models that can be embedded into the objects. This illustration serves the purpose of producing encapsulated behaviour descriptions for each object in the system from which typical sub-objects (that have to be encoded with PM knowledge) can be identified. This description in natural language of the imagery of the behaviour of a typical physical system should help in the formal definitions and the development and the constraining of a functional architecture for the system.

Consider the following example of a train of masses and springs, as shown in Figure 2. Mass *O4* is an object joined to a boundary by spring *R4* and the other masses are objects connected by springs as shown. Let t_x be a time interval for the message (wave) to take to reach *O2* from *O1*. At any given instance of time, all objects have attributes (or state variables) which describe the position of the object in space. If object *O1* is disturbed, the sequence of events can be reasoned qualitatively as follows using time steps of t_i as multiples of t_x . Consider that each mass and the attached spring is an object, in the form of a mental model with specified boundaries. The objects can behave either

- (1) due to an external disturbance from a meta system and/or
- (2) due to messages from nearest neighbours.

In-messages are messages received by each object and out-messages are messages sent by each object.

6.2.1. At time step t_1

O1 receives two in-messages. One is an external action (a disturbance or force) and the other is a null message from *O2* (since *O2* is passive). *O1* behaves (reacts to the force) and produces a new set of state variables (position, velocity and acceleration) and sends an out message (wave) to *O2*. The other objects remain passive.

6.2.2. At time step t_2

O1 continues to receive an in message from an external disturbance (a force) and a null message from *O2*, since the wave has not reached *O2*. *O2* receives two in messages, the out message from *O1* and the null message from *O3* sent out during time step t_1 . Both objects behave and send messages to their neighbours. Objects *O3*, *O4* remain passive.

At any time step t_n , each object receives messages from its neighbours, behaves, changes state and sends out messages to its neighbours which will be taken into account in the next time step. This process will continue until all the energy in the

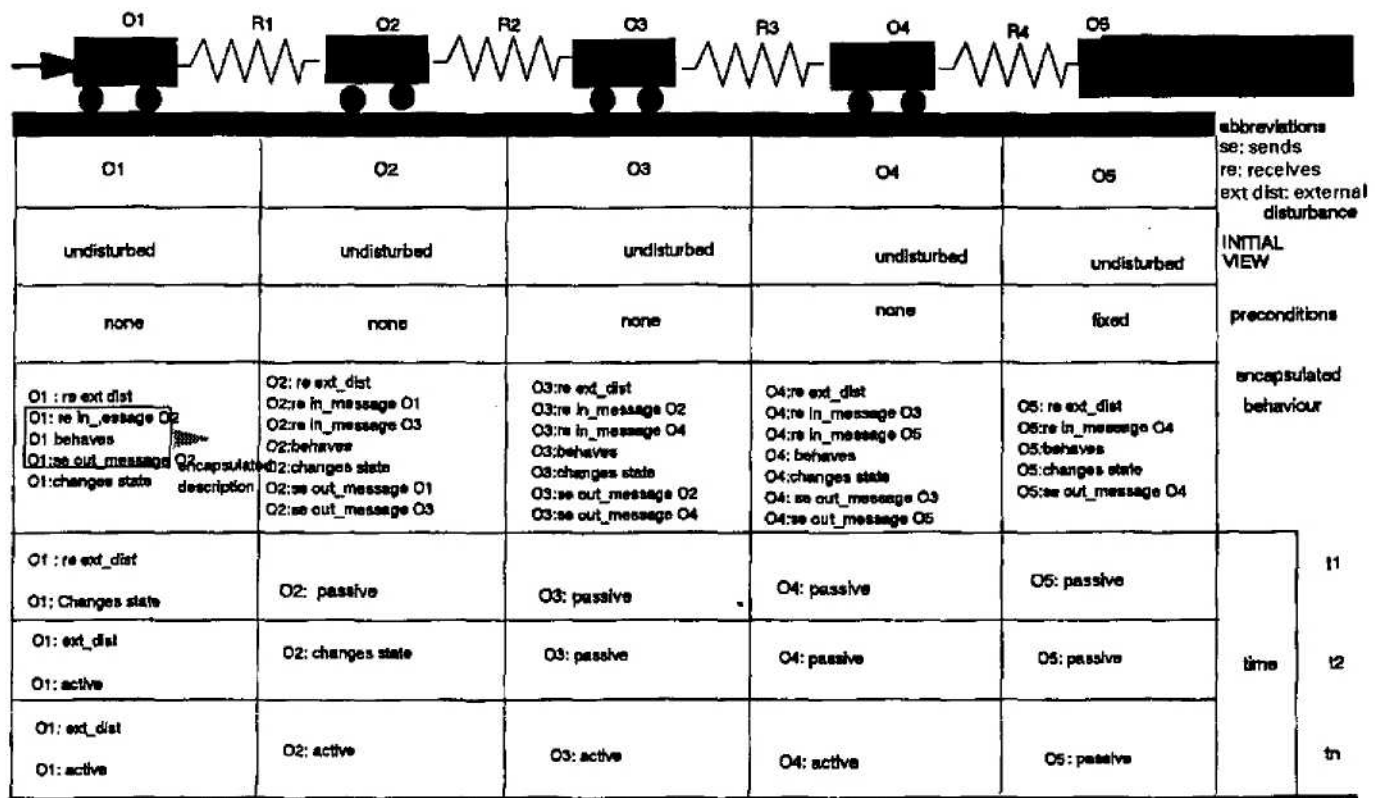


FIGURE 2. A qualitative model for a simple physical system.

system is dissipated. This model, to be called the IOPM, can be generalized in a formal language, that will facilitate the identification of objects that have to be encoded with deeper knowledge.

6.3. THE PHYSICAL PROCESS MODELLING LANGUAGE (PPML)

(1) The IOPM is a set of Interacting Objects, which interact by simultaneously communicating with neighbouring Interacting Objects. The process of communication and the resultant means of physical process representation is described by a PPML.

(2) An Object is a holon described in a (set of) meta languages and discriminated from its environment at an appropriate level of definition and from an appropriate perspective in order to solve a problem.

(3) A holon is an entity which is both a part and a whole. It is a part of a wider system (meta system) and is itself a system (of sub-systems).

(4) An Object is represented by a class which has characteristics. Characteristics are external (public) or internal (hidden). Characteristics may include attributes (features), behaviours (transformations) and constraints (conditions).

(5) Attributes (features) are predicates which define the state of an object. Attributes have names and values. Names remain static whereas the values of the attributes may change.

(6) A Behaviour is a procedure which changes the state of an object.

(7) A Constraint is a condition which defines the domain of behaviour of the object.

(8) An Interacting Object consists of sub-objects which are node objects, behaviour objects, and message objects.

(9) A neighbouring interacting object is an interacting object with which a given interacting object communicates by sending messages along a communication channel.

(10) A Node object represents a mass point in space and time. Node objects contain attributes inherited from an interacting object and neighbouring interacting objects.

(11) Attributes of an interacting object are its name, position in space, at a given instant of time, and relevant characteristics such as displacements, velocities and accelerations.

(12) A behaviour object is a class which contains procedures which change the state of an interacting object, according to physical laws. The behaviour object consists of sub-objects, which are external action objects, reaction objects and boundary objects.

(13) An external action object is a class which consists of procedures which input an impulse applied over a time interval, from a meta system. This inputs energy into the IOPM.

(14) A reaction object has procedures for the reactive behaviour in the interacting object. The reactive behaviour of an interacting object is dependent on the knowledge in the relation object. The reaction object is a constraint on the behaviour of the interacting object.

(15) A boundary object represents a sub-class of a behaviour object with attributes fixed in value, at a given instant in time, in the meta system. Boundary objects have knowledge that are constraints on the behaviour of an interacting object.

(16) A relation object is a class which contains specific knowledge for the reactive behaviour of an interacting object in a given situation. The knowledge is based on models of physical behaviour at an appropriate level of definition. It consists of sub-objects which define aspects of that behaviour. For an elastic solid media, these would be mass and stiffness objects.

(17) A message object is a class which contains an interface protocol for communication with neighbouring interacting objects.

(18) A message is a procedure which transmits the value of an attribute along a communication channel. A set of messages is a (set of) values(s) of the attributes of a interacting object. Messages are in the form of in-messages or out-messages.

(19) An in-message from a neighbouring interacting object, is processed by an interacting object and converted to an out-message by a behaviour. This causes the interacting object to change its state. The out-message is then sent to a (set of) neighbouring interacting objects.

(20) A communication channel is a degree of freedom of the interacting object.

(21) The degree of freedom of an interacting object is the capacity of the interacting object to permit transmission of the attributes in a defined coordinate direction separate and independent of other messages.

(22) The definition of a nearest neighbour for an interacting object is dependent on the spatial discretization of the physical process which the IOPM represents.

(23) A spatial discretization of the physical process is dependant on the transit time and the details of the physical process at an appropriate level of definition.

(24) The transit time is the time required to transmit a message requesting a change of state of an neighbouring interacting object. This time is dependent on the features of the physical properties and the knowledge for the reactive behaviour in the relation object.

(25) The transit time is the time taken by a disturbance in a physical process to travel the exact distance from one neighbour to the other and is also known as the critical time interval.

(26) For those IOPMs containing more than one type of disturbance, the critical time interval may be the minimum of the critical time intervals for the different types of disturbances occurring at that instant of time.

(27) A disturbance in a physical process is caused by external action.

(28) An event is the occurrence of a particular set of states for a set of objects at a given instant of time.

(29) A process is a set of temporally ordered discrete events separated by a time interval (step). Thus a process is a sequence of events.

(30) Every process is discrete and occurs in parallel.

(31) A process is passive if there are no changes in the values of the messages being passed between the objects in the process.

(32) A process is active when there is a change of state in any of the objects in the process.

(33) An IOPM which has reached the end of a process and is passive is in a state of static equilibrium.

(34) A passive process can be changed to a active process by an external action.

(35) A process stops when all of the external energy from the actions is dissipated by the IOPM.

- (36) Energy is dissipated in the IOPM by the behaviours of interacting objects.
- (37) A steady state process is a repeated active process, after a external action, if there is no dissipation of energy.
- (38) Energy dissipation is modelled at an appropriate level of definition, in the behaviour of an interacting object.
- (39) The IOPM is a closed world model at an appropriate level of definition.
- (40) The IOPM can be a open world model at an appropriate level of definition.

6.4 STRUCTURE OF THE IOPM

From the structure of the PPML, and the IOPM described through the simple physical system illustrated above, the structure of an interacting object can be developed.

6.4.1. Hierarchical structure of an interacting object

The lower level objects that can be encoded with knowledge are as follows.

(1) *Message object*

The message object is identified as an encoding that can receive messages (in-messages) from neighbouring interacting objects and send messages (out-messages) to neighbouring interacting objects. The messages in the present model contain the quantitative state vector. The vector is a message packet that contains attributes of the interacting object and its nearest neighbours.

(2) *Node object*

The node object is an internal memory of the interacting object with a reference system, which can transform local (internal) representations to a global representation that can be used by the message object to communicate with other interacting objects. The node object retains information of the interacting object's attributes and the attributes of its neighbours.

(3) *Behaviour object*

A behaviour object processes messages from neighbouring interacting objects, external action objects and relation objects with in the interacting object, based on physical laws or theories encoded in the behaviour object. A behaviour object computes a new set of state variables for the interacting object at every time step.

(4) *External action object*

The external action object effectively inputs energy into the system. It processes messages from a meta system, and consists of procedures which input impulses applied over a time interval into the behaviour object.

(5) *Reaction object*

A reaction object computes the reactive behaviour of an interacting object with the knowledge encoded in it. The reactive behaviour in solid mechanics may, for example, consist of the inertial force, damping force, and the restoring force which is from the stiffness of a spring or medium associated with the objects. A reaction object inherits knowledge from a relation object, which is encoded with a database of material and geometric response knowledge.

(6) *Boundary object*

A boundary object provides constraints on the behaviour of the system and has knowledge to separate the physical system from an external environment at an appropriate level of definition. For example in Figure 2 object *O4* is connected to a

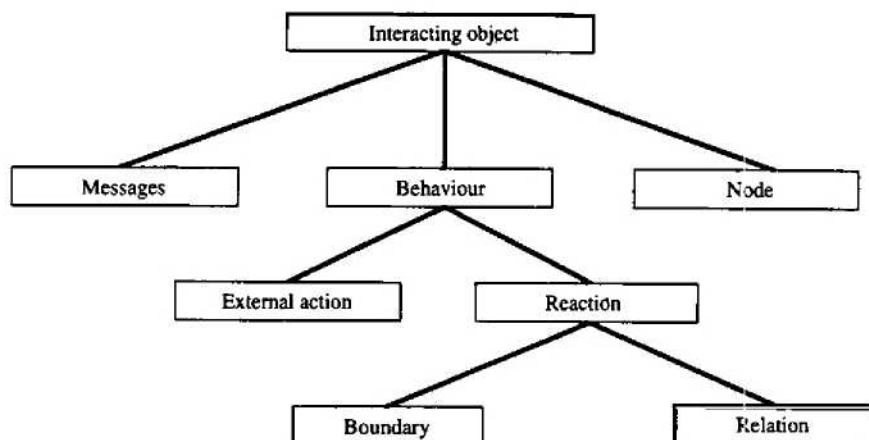


FIGURE 3. Conceptual structure of an interacting object.

boundary object *O5*. This object effectively has no reaction to the messages from *O4* and also transmits null messages. A boundary can be silent in certain channels and active in others. Knowledge of the boundary is encoded in the boundary object.

(7) *Relation object*

A relation object, as defined in the PPML contains knowledge of the behaviour of the intervening medium between two interacting objects and will represent the behaviour of springs in a mass-spring system or the intervening medium between two nodes in a continuum. This knowledge representation can be qualitative or mathematical based on physical laws.

The models embedded in the interacting objects (IOs) are described briefly in Section 7. The attributes of each IO have been chosen based on accepted spatial and temporal terminology in the form of displacements, velocities and accelerations. Kupiers (1984) and others for example use qualitative attributes to describe positional changes of physical objects. Here we presently use quantitative attributes although there is in principal no reason why any calculus should not be embedded into the objects. A typical structure of an IO is shown in Figure 3 although many variants will exist depending on the problem to be solved.

6.5. MESSAGE PASSING BETWEEN INTERACTING OBJECTS

The message passing protocol between interacting objects should enable communication between these objects even when they are placed on different processors in a concurrent computing environment. The IOPM has been designed such that the objects behave autonomously after receiving the messages. The message passing interface in the message object sends message packets which contain the attributes of an interacting object and received message packets which contain the attributes of the nearest neighbours. This concept has been implemented on a few parallel architectures. Typically, communication between interacting objects is as shown in Figure 4. The development of the mathematical model and implementation on parallel computers is discussed in detail in Chandra (1992).

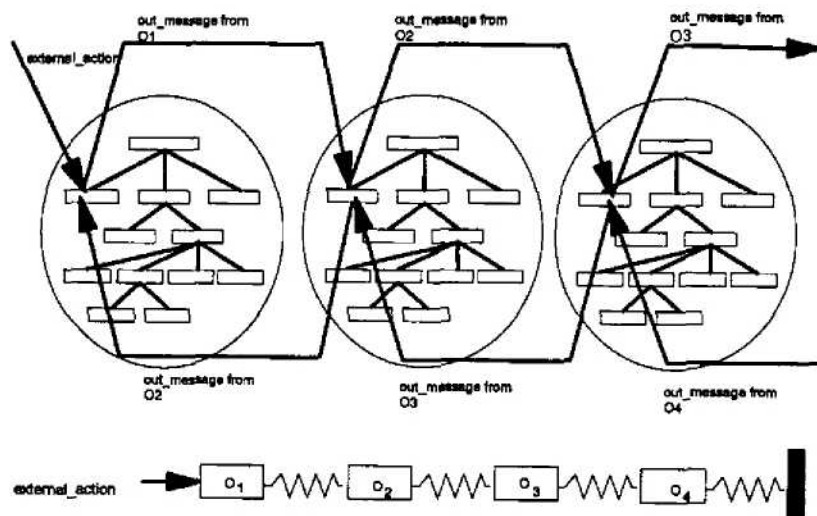


FIGURE 4. Message passing between interacting objects.

6.6. THE PHYSICAL LAWS

The reactive behaviour of common physical systems can be expressed using Newton's third law of action and reaction. The reactive behaviour can itself be identified as inertial force (from Newton's first law), the damping force which represents the dissipation of energy, and the restoring force, which is the reaction of the interacting object along with the medium associated with it. The behaviour is represented as follows:

```

interacting object::behaviour
{
  for all nearest neighbours: receive in message
  external action
  reaction
  acceleration = (external action-reaction)/mass
  change state {
    new position = old position + displacement
  }
  for all nearest neighbours: sent out_message

```

The reaction object is encoded as

```

object::reaction
{
  damping force
  restoring force
}

```

7. The mathematical model

Penrose (1989) describes mathematical modelling as a spontaneous and mysterious process. Aside from the controversy associated with beliefs in strong AI or in the

extra-ordinary connection between mathematics and reality, it is a thesis of this paper that early qualitative models can be evolved into mathematical models which can be represented on computers.

The mathematical model that has to be encoded as knowledge in the interacting object has to use the information that is received by the message object of the interacting object (in the form of attributes) and process it according to the physical laws encoded in the mathematical model. A model currently implemented in the IOPM are equations of motion:

$$m\ddot{x} + c\dot{x} + kx = f \quad (1)$$

where \ddot{x} is the acceleration, \dot{x} the velocity, and x the displacement of a particle or object at a given instant of time. In the usual notation $m\ddot{x}$ is the inertia, $c\dot{x}$ is the damping force, and kx the restoring force for a given particle.

Consider a typical physical system as shown in Figure 2. Let each mass be represented by an interacting object O_i which can be regarded as a computer data structure encoded with knowledge. The interacting objects have the following attributes (with respect to position and motion) at a given instant of time.

displacement $x_{i,t}$

velocity $\dot{x}_{i,t}$

acceleration $\ddot{x}_{i,t}$

Any object O_i has two nearest neighbours O_{i-1} and O_{i+1} in the system. The behaviour of the object O_i is influenced by the springs between O_i and its neighbours. At any instant of time, the position of O_{i-1} and O_{i+1} will be required to compute the restoring of the interacting object O_i . F_i is the force from an external disturbance that is applied at a given instant of time to the object O_i and m_i is the mass, k_{i-1} is the stiffness of the spring with respect to its neighbour O_{i-1} , k_{i+1} is the stiffness of the spring with respect to its neighbour O_{i+1} , c_i is the viscous damping constant and the damping force is assumed to depend on the velocity (Timoshenko, 1974).

The equation of motion for any object O_i can be written as

$$m_i\ddot{x}_i + c\dot{x}_i + k_{i-1}(x_i - x_{i-1}) + k_{i+1}(x_i - x_{i+1}) + F_i = 0.2 \quad (2)$$

where x_{i-1} and x_{i+1} are the displacements of the neighbours at the previous time-step. The equation of motion to find out a change of state of any object O_i at the end of a time-step is required. The well known relationship between displacement, velocity and acceleration is given by:

$$x_{i,t+1} = x_{i,t} + \Delta t\dot{x}_{i,t} + 0.5\Delta t^2\ddot{x}_{i,t} \quad (3)$$

$$\dot{x}_{i,t+1} = \dot{x}_{i,t} + \Delta t\ddot{x}_{i,t} \quad (4)$$

The accelerations $\ddot{x}_{i,t+1}$ at the next instant of time is given by the recursive relation:

$$m\ddot{x}_{i,t+1} = f_i - c_i\dot{x}_i - k_ix_i \quad (5)$$

This set of equations are the typical behaviour models encoded in the behaviour object and sub-objects that are inherited by the behaviour object as discussed in detail in Chandra (1992).

8. IOPM on parallel computers

Most parallel computer models to date have been developed by exploiting the algorithmic or task parallelism in already developed mathematical models. However, if models of parallelism are to mimic the contention that nature is parallel, they should be designed in a constraining functional architecture at the cognitive level, that can be carried down to the programming level in a consistent manner.

Physical processes are naturally parallel in that there is always a certain number of events that are happening simultaneously. Natural parallelism is identified in the IOPM by localising events so that several independent concurrent processes are naturally built in. Until parallel computing machines became available, these processes were modelled using sequential machines. Even with the impressive speed of the current generation of computers many problems in physical process modelling still remain intractable. Parallel computing has made some of these problems tractable.

The three major commercial architectures currently available implement parallelism differently. These are:

- (1) Multi-computer (MIMD, distributed memory): data and processing distributed,
- (2) shared memory MIMD: only the processing is distributed, and the user does not need to make explicit the underlying data parallelism, which leads to the parallel processing,
- (3) distributed memory SIMD: data and synchronous processing distributed.

Ideally, to take advantage of the natural parallelism of the IOPM, there should be an individual processing element in a parallel computer for each process. A typical architecture is proposed in Figure 5. Utilities are built for implementing the software on various types of parallel hardware. Where the number of interacting objects exceeds the number of processors the interacting objects are allocated to a processor. The structure of the programme does not change even though implementation is heavily hardware dependent because message passing takes place within a

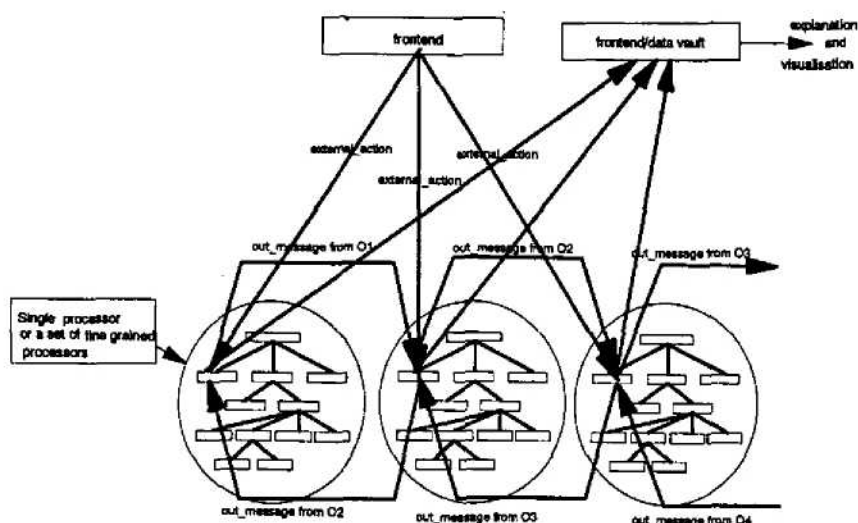


FIGURE 5. Communication and processing architecture for the IOPM.

processor if a number of interacting objects are on the same processor. Otherwise the message packet may be sent to another processor via a host or directly depending on the topology. Some central control is required which is necessary to monitor the simulation from a front end.

The operation of the IOPM on a transputer based (MIMD) parallel computer and a massively parallel (SIMD) Connection Machine will now be outlined. More details are available in Chandra (1992). The transputer is a VLSI design based processor that has a number of bi-directional links for communication and the topology can be arranged in an order that will produce the maximum efficiencies. The Meiko Computing Surface (MCS), a transputer based parallel computer was used. On the transputer based hardware, MCS provides an operating system and tools (CS Tools) in UNIX which can be configured to suit our requirement. CS Tools uses a communicating process approach of organizing data to cooperate on the performance of a single task. The computer available was a 24 processor machine.

The program implementation for the IOPM was carried out as conceived at the primary stage. A utility was designed to route the message packets through the host processor to different processors. Since a large number of processors were not available in the transputer based machine, a set of interacting objects were deployed on each of the processors. This meant that message packets containing the attributes of a set of nearest neighbour interacting objects were sent across to other processors, if they did not exist on a single processor. In other cases, the exchange of information (attributes) was within the processor itself.

The Connection Machine is basically a SIMD machine with a very large number of small processors. A fully configured CM-2 can have 64 000 processors and is capable of general and grid communication. The grid communication system is more structured and efficient. Using C*, a parallel implementation of C enables a declaration of parallel variables that can be placed on each processor. Each interacting object with its attributes was designed as a nested C structure and was placed on separate processors. If the size of the problem is large, the concept of a virtual processor is utilized. The SIMD architecture enables access to attributes of all the interacting objects at the same time and as a result explicit user message passing was unnecessary in the general communication approach.

9. An appropriate computational environment for description, simulation and explanation

A single unifying framework and computational environment to provide a wide variety of modelling concepts, relationships and techniques for description, simulation and explanation of physical processes is desirable. It is important in solving a given problem, to be able to choose models that represent processes adequately with efficient utilization of resources. Inherent in the building of this framework is the belief a concept of appropriate physics can be employed. The term refers to the selection of a level of definition for the problem on hand, that is dependent on the knowledge and resources available to solve the problem. Using the IOPM a flexible frame work can be generated to accommodate objects that can be encoded with different kinds of knowledge.

In the development of the IOPM, some progress has been made in identifying

building blocks which can be encoded with appropriate knowledge either from qualitative physics or mathematical models. Further identification and construction of such functional objects will allow a hierarchy of possibilities and modelling of relationships between differing levels of definition. In this work the possibility has been explored of providing such a framework to represent simple physical processes in solid mechanics which could be extended to other models.

In order to be able to assimilate the results of a simulation easily some form of qualitative explanation is required. A procedure for mapping quantitative attributes of the physical system into qualitative attributes and operating with them is described in Chandra, Blockley and Woodman (1993). An IOPM simulation

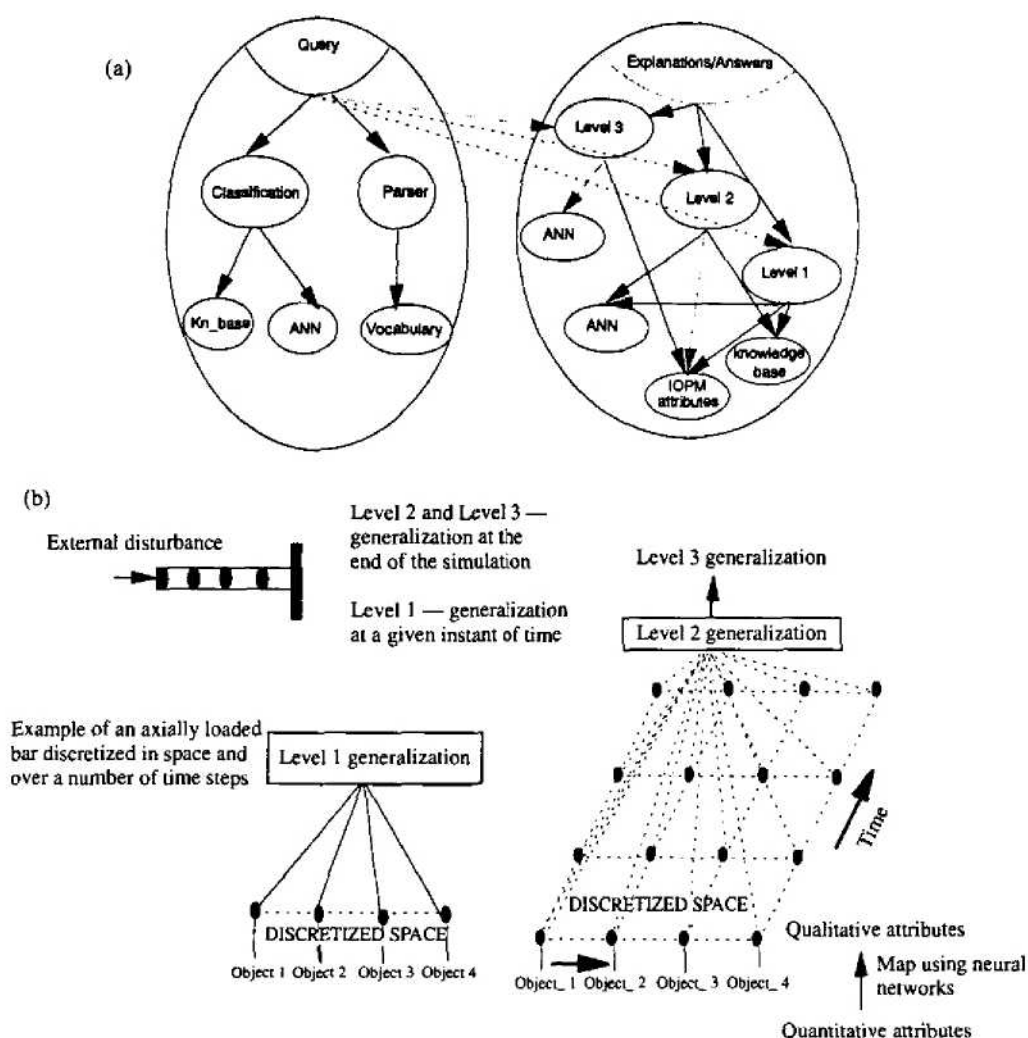


FIGURE 6. (a) Structure of the explanation system. —: Inheritance; ----: messages; ANN: artificial neural networks. (b) Levels of generalization adopted.

produces a vast amount of quantitative data. The quantitative attributes are converted into a set of qualitative attributes in the form of, for example, values of low, high, medium etc. Neural networks are used to compress these attributes into generalizations which can be regarded as explanations for queries. The system has been designed such that Natural language and the PPML is part of the vocabulary of the system. Figures 6(a) and (b) show the basic structure of the system.

10. Discussion

A structured methodology to build cognitive models into computational models has been demonstrated. Naturally parallel modelling of a physical system has been used to illustrate the procedure proposed. This is done by designing a functional architecture for the model through natural language description defining a formal language, developing functional objects, identifying physical laws and mathematical models which are embedded in the objects and parallel computer implementations. It is believed that such a hierarchical and structured methodology can aid efficient and stable process modelling development. A relatively simple procedure has been proposed to document the development of early models, which we hope can evolve into a more formal procedure on the accumulation of empirical evidence, if found successful in other systems. Such experiences we believe may, provide some understanding of how primitive models are generated.

A practical structure has been provided for a complete system that can be used to describe, simulate and explain the behaviour of a physical system. In particular the possibility has been explored for providing qualitative explanations from simulations that produce vast amounts of quantitative data, which is easy to assimilate. The model uses established physical laws and mathematical models that have been falsified by experiment and engineering judgement. Non quantitative concepts and relations, such as those developed in qualitative physics and reasoning, could also be embedded as qualitative knowledge into the objects in further development of these ideas.

11. Conclusion

A natural model for the concept of appropriate physics is a set of autonomous objects encoded with knowledge, which can be used as building blocks that will provide a choice of modelling concepts and relationships. This will allow a hierarchy of possibilities and modelling of relationships between differing levels of definition. This paper has explored the possibility of such a framework to represent simple physical processes in solid mechanics, which can be extended to other models. The IOPM can be used to demonstrate the development of cognitive models, using functional architecture to constrain the model. It provides a suitable language, for identifying physical laws, mathematical representations and computer models for parallel computers. It is believed that such a hierarchical and structured methodology can help in the development of efficient and stable process models.

References

- BURTON, R. (1968). *Vibration and Impact*. London: Dover Publications.
- BLOCKLEY, D. I. (1980). *The Nature of Structural Design and Safety*. Hemel Hempstead: Ellis Horwood.
- CHANDRA, S. (1992). *Knowledge based physical process modelling and explanation*. Ph.D thesis, University of Bristol, Bristol.
- CHANDRA, S., BLOCKLEY, D. I. & WOODMAN, N. J. (1993). Qualitative querying of physical process simulations. *Civil Engineering Systems*, **10**, 225-242.
- CONRAD, M., HOME, D. & JOSEPHSON, B. (1988). Microphysical reality and quantum formalism. In G. TAROZZI, A. VAN MERWE & F. SELLER], Eds. pp. 285-293. Boston, MA: Kluwer Academic.
- DE KLEER, J. & BROWN, J. S. (1984). Commonsense reasoning about causality: deriving behaviour from structure. *Artificial Intelligence*, **24**, 169-203.
- FEYNMAN, R. (1965). *The Feynman Lectures on Physics*, vol III. Reading, MA: Addison-Wesley.
- FORRUS, K. D. (1984). Qualitative process theory. *Artificial Intelligence*, **24**, 85-168.
- FORBUS, K. D. & FLANKENHAINER, B. (1990). Self explanatory simulations: an integration of qualitative and quantitative knowledge. *Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90)*, pp. 380-387, Cambridge, MA: MIT Press.
- FOX, G. C. (1989). Parallel computing comes of age: super computer level parallel computations at Caltech. *Concurrency: Practice and Experience*, **1**, 63-103.
- GENTNER, D. & STEVENS, A. L. (1983). *Mental Models*. Hillsdale, NJ: Lawrence Erlbaum.
- HAYES, J. R. & SIMON, H. A. (1976). The understanding process: problem isomorphs. *Cognitive Psychology*, **8**, 165-180.
- KULKARNI, D. & SIMON, H. A. (1988). The process of scientific discovery. *Cognitive Science*, **12**, 139-175.
- KUPIERS, B. (1984). Commonsense reasoning about causality: deriving behaviour from structure. *Artificial Intelligence*, **24**, 169-203.
- LEITCH, R. R., WEIGAND, H. E. & QUEK, H. C. (1990). Coping with complexity in physical system modelling. *AICOM*, **3**, 48-53.
- LINDSAY, R. B. (1968). *The Nature of Physics*. London: Brown University Press.
- NORMAN, D. A. (1983). Some observations on mental models. In D. GENTNER and A. L. STEVENS, Eds. *Mental Models*, Hillsdale, NJ: Lawrence Erlbaum.
- PENROSE, R. (1989). *The Emperor's New Mind: Concerning Computers, Minds and the Laws of Physics*. Oxford: Oxford University Press.
- PYLYSHYN, Z. W. (1989). *Computation and Cognition: Towards a Foundation for Cognitive Science*. New York, NY: MIT Press.
- ROSEN, R. (1985). *Anticipatory Systems*. Oxford: Pergamon Press.
- STAGGERS, N. & NoRcio, A. F. (1993). Mental models: concepts for human-computer interaction research. *International Journal of Man-Machine Studies*, **38**, 587-605.
- TIMOSHENKO, S., YOUNG, D. H. & WEAVER Jr, W. (1974) *Vibration Problems in Engineering*. New York, NY: John Wiley.
- TRAINOR, L. E. H. & WISE, M. B. (1979). From physical concepts to mathematical structure: an introduction to theoretical physics. Toronto: University of Toronto Press.
- WILLIAMS, M. D., HOLLAN, J. D. & STEVENS, A. L. (1983). Human reasoning about physical systems. In D. GENTNER & A. L. STEVENS, Eds. *Mental Models*. Hillsdale, NJ: Lawrence Erlbaum.
- ZAIONC, A. G. (1976). Goethe's theory of colour and scientific intuition. *American Journal of Physics*, **44**, 327-333.